Extrinsic Calibration Algorithm between a Stereo Visual System and a 3D LiDAR

Yang Zhou[1]

Abstract— **Camera and light detection and ranging (LiDAR) are frequently used for perception in real-world applications. The combination of these heterogeneous sensors bring advantages of both kinds of sensors and get accurate and promising results. The extrinsic calibration of camera and LiDAR system is a prerequisite for robot perception applications, the robustness and usability are required in research and industry. Previous works have provided several extrinsic calibration methods including target-based works and target-less works. Although there are many works dealing with calibration of a monocular camera and a 3D LiDAR, the solutions of the calibration of the stereo visual system and 3D LiDAR still have not meet the requirement of the real-world application yet. In this work, we designed an extrinsic calibration algorithm which can be used for calibration of a stereo visual system and a 3D LiDAR system. Our method can estimate extrinsic parameters between a stereo visual system and a 3D LiDAR with only one checkerboard pose. We can further improve the result by a joint non-linear optimization considering LiDAR disparity constraint using multiple poses.**

I. Introduction

Nowadays, the number of different sensors mounted on robots is increasing. Different sensors have different characteristics in real-world robotics applications which enable robots to perceive the environment in challenging situations. To utilize the advantages of different sensors, the fusion system of different modalities have been widely applied to perception, navigation, and mapping applications. The transformation relationship between different sensors is required for sensor fusion system to align information in a common coordinate system. Since robot perception applications are highly relying on the intrinsic parameters and extrinsic parameters of different sensors, the calibration of multiple heterogeneous sensors needs to be precise. The calibration method needs to be robust for different settings and needs to be convenient and user-friendly.

LiDAR and stereo visual system have different characteristics. Light detection and ranging (LiDAR) is a range sensor can obtain accurate range measurements using laser beams. 3D LiDAR uses multiple laser beams to produce precise 3D point cloud, so it can sense the geometry feature of the environment. Visual Camera can obtain color and texture information of the scene. The stereo visual system has two cameras which can capture 3D information of scene based on feature matching between two cameras. To utilize the advantages of the stereo visual system and the 3D LiDAR, we need to calibrate these two different sensors.

The research focusing on this problem, however, is limited. Although there are some works focusing on the extrinsic calibration of 2D LiDAR and stereo visual system and some works focusing on the calibration of 3D LiDAR and monocular camera, there are very few works dealing with extrinsic calibration between a stereo visual system and 3D LiDAR. Among these works, some methods [1], [2], [3], [4] exploit mutual information between different sensors. Some methods [5], [6], [7] use special designed calibration target. And some works [8], [9], [10], [11] use a checkerboard which is common to users as the calibration target. We also choose checkerboard as our calibration target because it enables us to calibrate the intrinsic and extrinsic parameters of the stereo visual system and LiDAR simultaneously.

To tackle the extrinsic calibration problem, we need to find constraints between the stereo visual system and 3D LiDAR to find the geometry relationship between these two different sensors. The plane of checkerboard has been used as geometry constrained in the extrinsic calibration of a 2D LiDAR and a camera. They calculate the extrinsic parameters only by plane correspondences which require 3 poses as a minimum number. Line correspondences and point correspondences are also explored in many works on the extrinsic calibration of a 2D LiDAR and a camera. To reduce the minimum number requirement of poses, we obtain 3D line and plane correspondences according to 4 boundaries of checkerboard and 1 plane of the checkerboard. We can estimate 2D line and 3D plane feature on the image based on the line detection algorithm and intrinsic parameters of the camera. 3D line feature on the image is calculated by the intersection of the back-projected plane of 2D line extracted from the image and the 3D plane of the checkerboard extracted from the feature points on the checkerboard and the intrinsic parameters of the camera. We can estimate 3D line and 3D plane feature based on LiDAR point cloud using the RANSAC[12] algorithm to eliminate outliers. Using 1 plane correspondence and 2 of 4 line correspondences, we can estimate extrinsic parameters of a monocular camera and a 3D LiDAR with one pose using a close-formed solution. After calibrating each single cameras with 3D LiDAR, we globally optimize extrinsic parameters with LiDAR disparity constraint by non-linear optimization.

The contribution of this paper can be summarized as:

1) Extend extrinsic calibration of a monocular camera and a 3D LiDAR [13]to extrinsic calibration of a stereo visual system and a 3D LiDAR.

[1]Yang Zhou is with School of Information Science and Technology, ShanghaiTech University, Shanghai, China. This work is done during Robotics Institute Summer Scholar program in the Robotics Institute, Carnegie Mellon University. zhouyang@shanghaitech.edu.cn

2) We reduce the minimum requirement of poses from 3 to 1 by utilizing 3D correspondences of lines and planes. The system can be extended to extrinsic calibration of multi-camera and multi-LiDAR system.
3) We introduce LiDAR disparity constraint to improve the result with multiple poses by non-linear optimization.
4) We develope an extrinsic calibration software in C++ with high efficiency and robustness.

II. Related Works

There are different methods purposed to solve multiple sensors calibration problem, here we will talk about related works according to different problem categories about the camera to LiDAR calibration.

For the monocular camera to LiDAR calibration problem, there are different types of approaches based on different correspondences. Some methods do not rely on the calibration target. In [14] and [4], they make use of mutual information between LiDAR reflectivity and camera intensity to do the extrinsic calibration outdoors without a calibration target. In [15] relies on PnP algorithm [16] using manually selected points correspondences. In [17], the deep neural network is applied on extrinsic calibration to solve this problem by end-to-end training.

Some methods use a rectangle calibration target such as a checkerboard. [8] uses a checkerboard to do extrinsic calibration between a perspective camera and a 2D LiDAR, they use the plane-line correspondence established by LiDAR points on plane and plane parameters estimated in the camera coordinate system, their algorithm requires at least 5 poses to get the extrinsic parameters. [9] use plane-plane correspondence established by estimating plane parameters from 3D LiDAR and camera to estimate initial rotation matrix and translation vector, then they use non-linear optimization to refine the result by minimizing point to plane distance. Their method needs at least 3 poses. Instead of using one checkerboard, [10] uses several checkerboards in front of sensors to do the extrinsic calibration with one pose, which avoids moving checkerboards. Methods based on plane constraint can be easily extended to the multi-sensors system, [11] calibrate the extrinsic parameters between an omnidirectional camera and a 3D LiDAR. When estimating plane parameters from the image and 3D LiDAR, a plane from a farther distance cannot be estimated as precisely as the plane from a closer distance. [13] can do extrinsic calibration between a camera and 3D LiDAR using lines and planes correspondences in one pose.

Some methods rely on specifically designed calibration target. In [5], arbitrary trihedron is used to estimate the extrinsic parameters with 2 poses. [6] uses a discontiguous calibration target to emphasize 2D LiDAR information to estimate extrinsic parameters between a camera and a 2D LiDAR. In [7], the v-shaped target is used to utilize the boundary of the target captured by LiDAR. These works use the target with a specially designed pattern to exploit the boundary information. The boundary information can be detected by LiDAR and visual system as correspondences, which can reduce the number of poses required for extrinsic calibration.

For the stereo camera to LiDAR calibration problem, [18] calibrate a 3D LiDAR and a stereo visual system using inertial measurement unit (IMU). With the help of inertial data, they used a bright spot as the only calibration target. Their framework can be extended to the multi-camera network. In [19], a board with circular hole pattern is used to calibrate a 3D LiDAR and a stereo visual system. The circular hole pattern can be detected in camera frame and LiDAR frame robustly. In [20], the paper presents an extrinsic calibration algorithm between a stereo vision system and a 2D LiDAR-based on the 3D reconstruction of the checkerboard. They use 3D corner points of checkerboard obtained by stereo camera system to do triangulation, solve least-square estimation of the 3D plane of checkerboard and use non-linear optimization to optimize extrinsic parameters. [21] used the particle swarm optimization algorithm (PSO) to estimate extrinsic parameters and fuse information of a stereo camera and a LiDAR without the aid of another calibration target. [22] proposed a method to calibrate a 2D LiDAR and a multi-camera system, it decouples the problem into two hierarchical level optimization problem without using any calibration target.

Comparing to the above works, our method utilizes 3D planes and lines correspondences between camera and LiDAR, we use checkerboard to extract the plane and boundary of checkerboard both from camera and LiDAR. Considering there will be some cases that intrinsic parameters of the camera are unknown, the checkerboard can be used to estimate intrinsic parameters at the same time. The boundary of the checkerboard can be extracted easily because of the visual feature and reflectance feature of the checkerboard. To consider the stereo visual system and 3D LiDAR globally, our method refines the initial optimization result according to the geometry constraint of the stereo visual system which leads to the better result.

III. Methods

A. Problem Formulation

This section will describe the formulation of the extrinsic calibration problem of a stereo visual system and a 3D LiDAR. The extrinsic calibration of a stereo visual system and a 3D LiDAR is to estimate the relative rotation and translation between two cameras and the 3D LiDAR. The relationship of the three sensors is shown in Fig. 1. We use $\mathbb{R}^3_{C_1}$ as the coordinate system of the left camera, $\mathbb{R}^3_{C_2}$ as the coordinate system of the right camera, and $\mathbb{R}^3_l$ as the coordinate system of the 3D LiDAR.

For camera system, we will do intrinsic calibration to both cameras to get the intrinsic parameters of cameras and undistort image by estimated distortion coefficients by the method described in [8] Then we can model both cameras by pinhole camera model. We denote $\mathbf{K}^{C_1}$ as the intrinsic matrix of the left camera and $\mathbf{K}^{C_2}$ as the intrinsic matrix of the right camera. The points in left camera coordinate

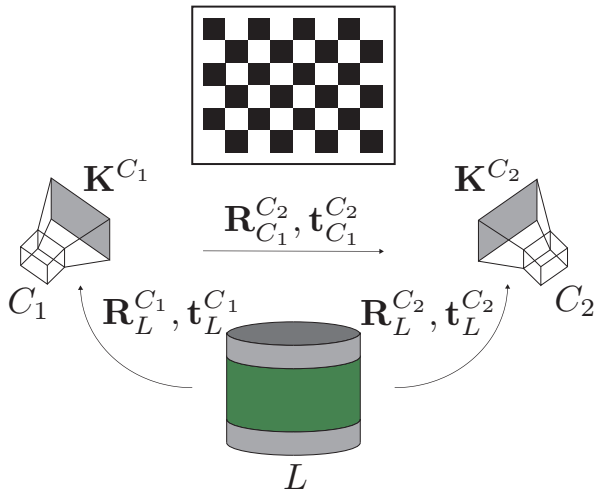

Fig. 1. Sensors Relashionship

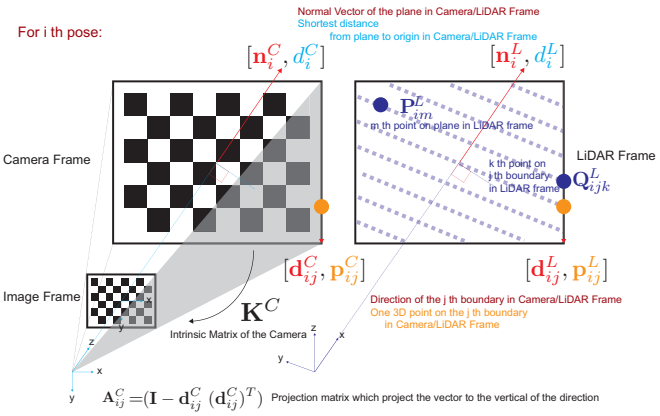

Fig. 2. Notation of plane and line parameters

system are presented as $\mathbf{P}^{C_1} : (X^{C_1}, Y^{C_1}, Z^{C_1}) \in \mathbb{R}^3_{C_1}$, and points in right camera coordinate system are presented as $\mathbf{P}^{C_2} : (X^{C_2}, Y^{C_2}, Z^{C_2}) \in \mathbb{R}^3_{C_2}$. The pixels of points in left camera image can be described as (x^{C_1}, y^{C_1}). The pixels of points in right camera image can be described as (x^{C_2}, y^{C_2}). The relationship between the camera coordinate system and camera image frame can be described as below.

$$\mathbf{K}^{C_1}\mathbf{P}^{C_1} = \begin{bmatrix} x^{C_1} \\ y^{C_1} \\ 1 \end{bmatrix} Z^{C_1} \tag{1}$$

$$\mathbf{K}^{C_2}\mathbf{P}^{C_2} = \begin{bmatrix} x^{C_2} \\ y^{C_2} \\ 1 \end{bmatrix} Z^{C_2} \tag{2}$$

For LiDAR system, we convert the distance from the direction of each scan points in polar coordinates obtained by 3D LiDAR to the form of Cartesian coordinates as $\mathbf{P}^L \in \mathbb{R}^3_l$.

For extrinsic parameters, we decouple the transformation relationship between cameras and 3D LiDAR as rotation matrix $\mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2} \in SO(3)$ and $\mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2} \in \mathbb{R}^3$. Let $(\mathbf{R}_L^{C_1}, \mathbf{t}_L^{C_1})$ and $(\mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_2})$ be the relative rotation matrix and translation vector of left camera to 3D LiDAR and right camera to 3D LiDAR. Let $(\mathbf{R}_{C_1}^{C_2}, \mathbf{t}_{C_1}^{C_2})$ be the relative rotation matrix and translation vector from the left camera to the right camera. The relationship of three sensors can be described as below:

$$\mathbf{P}^{C_1} = \mathbf{R}_L^{C_1}\mathbf{P}^L + \mathbf{t}_L^{C_1} \tag{3}$$

$$\mathbf{P}^{C_2} = \mathbf{R}_L^{C_2}\mathbf{P}^L + \mathbf{t}_L^{C_2} \tag{4}$$

$$\mathbf{P}^{C_2} = \mathbf{R}_{C_1}^{C_2}\mathbf{P}^{C_1} + \mathbf{t}_{C_1}^{C_2} \tag{5}$$

For ith pose, we extract 3D plane features from both cameras and 3D LiDAR. The 3D plane parameters in cameras are described as $[\mathbf{n}_i^{C_1}; d_i^{C_1}]$ and $[\mathbf{n}_i^{C_2}; d_i^{C_2}]$. $\mathbf{n}_i^{C_1}, \mathbf{n}_i^{C_2}$ represent the normal vector of the plane in the camera frame, $d_i^{C_1}, d_i^{C_2}$ represent the distance from the plane to the origin of the camera coordinate system. The 3D plane parameters in LiDAR are described as $[\mathbf{n}_i^L; d_i^L]$. $\mathbf{n}_i^L$ represents the normal vector of the plane in the LiDAR frame, d_i^L represents the distance from the plane to the origin of the LiDAR coordinate system. For any point $(X_1, Y_1, Z_1) \in \mathbb{R}^3_{C_1}$ that lie on plane i in left camera , any point $(X_2, Y_2, Z_2) \in \mathbb{R}^3_{C_2}$ that lie on plane i in right camera, point $(X_l, Y_l, Z_l) \in \mathbb{R}^3_l$ that is the centroid point of plane i in 3D LiDAR,

$$[\mathbf{n}_i^{C_1}, d_i^{C_1}] \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ -1 \end{bmatrix} = \mathbf{0} \tag{6}$$

$$[\mathbf{n}_i^{C_2}, d_i^{C_2}] \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \\ -1 \end{bmatrix} = \mathbf{0} \tag{7}$$

$$[\mathbf{n}_i^l, d_i^l] \begin{bmatrix} X_l \\ Y_l \\ Z_l \\ -1 \end{bmatrix} = \mathbf{0} \tag{8}$$

For the ith pose, we also extract 3D line features from both cameras and 3D LiDAR. The the jth 2D line in left camera and right camera are described as $l_{ij}^{C_1}$, $l_{ij}^{C_2}$ which contains image-space coordinates of starting points and ending points. The the jth 3D plane in left camera and right camera are denoted as $[\mathbf{d}_{ij}^{C_1}, \mathbf{p}_{ij}^{C_1}]$ and $[\mathbf{d}_{ij}^{C_2}, \mathbf{p}_{ij}^{C_2}]$, where $\mathbf{d}_{ij}^{C_1}, \mathbf{d}_{ij}^{C2} \in \mathbb{R}^3$ are derections of 3D lines in both cameras and $\mathbf{p}_{ij}^{C_1}, \mathbf{p}_{ij}^{C2} \in \mathbb{R}^3$ are points lie on 3D lines in both cameras. We use $\mathbf{A}_{ij}^{C_1}$ and $\mathbf{A}_{ij}^{C_2}$ to represent the projection matrix $(\mathbf{I} - \mathbf{d}_{ij}^{C_1}(\mathbf{d}_{ij}^{C_1})^T)$ and $(\mathbf{I} - \mathbf{d}_{ij}^{C_2}(\mathbf{d}_{ij}^{C_2})^T)$.

The jth 3D line in 3D LiDAR can be described as $[\mathbf{d}_{ij}^L, \mathbf{p}_{ij}^L]$, where $\mathbf{d}_{ij}^L$ is the direction of the 3D line and $\mathbf{p}_{ij}^L$ is the centroid of points lie on the 3D line. We denote the kth point lie on the jth 3D line in 3D liDAR as $\mathbf{Q}_{ijk}^L$, and the mth point lie on the jth 3D plane in 3D LiDAR as $\mathbf{P}_{im}^L$. In ith pose, the number of points on the plane is denoted as N_i, the number of points on the jth boundary is denoted as K_{ij}.

In this paper, $L_{ij}^{C_1}$, $L_{ij}^{C_2}$ and L_{ij}^{L} represent the jth line in the ith pose of two cameras and 3D LiDAR, $\pi_i^{C_1}$, $\pi_i^{C_2}$ and π_i^{L} represent plane in the ith pose. The plane and line parameters are also illustrated in Fig. 2. We use $\cdot$ to represent the dot product.

B. System Description

We implement the extrinsic calibration algorithm of the stereo visual system and the 3D liDAR in C++. We will discuss the structure of the calibration software. The basic structure of the system is shown in [Fig].

To use this software, users need to use ROS [23] to record several bags containing images from both cameras and point clouds from 3D LiDAR. If intrinsic parameters of both cameras are not provided, users need to place the checkerboard in different positions to make sure that all image area is covered by different positions to get accurate intrinsic parameters. The point cloud and images will also be used to calibrate the extrinsic parameters. Users record images and point clouds for several seconds in rosbag at each position. Users need to provide a cuboid in LiDAR coordinate system which contains the checkerboard to help locate the checkerboard in the point cloud. Then this information are regarded as the input of the system.

Firstly, the system will use intrinsic calibration [8] built-in OpenCV [24] to calibrate the intrinsic parameters of both cameras using images containing checkerboard, then undistorted images will be stored in a folder for further processing.

In the second step, stereo calibration in OpenCV will be used to obtain initial extrinsic parameters of both cameras.

In the third step, 3D planes and lines feature from both cameras and 3D LiDAR are extracted. LSD method [8] is used to extract 2D lines from images, and plane parameters in cameras are estimated using checkerboard features. 3D planes in 3D liDAR are estimated by RANSAC [25], the 3D lines in 3D planes in LiDAR frame are also estimated by RANSAC after denoising.

In the fourth step, we use the SVD method to give out the initial guess of extrinsic parameters between each camera and the 3D liDAR.

In the fifth step, we utilize the LiDAR disparity constraint to refine the extrinsic calibration result globally by Levenberg-Marquardt algorithm[26].

C. Feature Extraction of Calibration Target

In this section, we will introduce the steps of feature extraction in detail.

1) Planes and Lines Extraction from Camera: We will consider feature extraction in cameras, without loss of generality, we will discuss it in the left camera.

The 3D plane parameters $[\mathbf{n}_i^{C_1}, d_i^{C_1}]$ can be computed by homography using checkerboard. After extracted all 2D lines by LSD[27] method, we can extract several line segments on the four boundaries of the checkerboard. After line fusion, we can get starting points and ending points $[(x_s, y_s), (x_e, y_e)]$ of a line. A sample result of 2D boundary detection is shown in Fig. 3. We can get back-projected 3D plane of the line which is the plane containing the 3D line and origin of $\mathbb{R}_{C_1}$ as $[K_{C_1} \cdot ([x_s, y_s, 1]^T \times [x_s, y_s, 1]^T); 0]$. Then we can get the 3D plane parameters by the intersection of the back-projected 3D plane and the 3D plane.

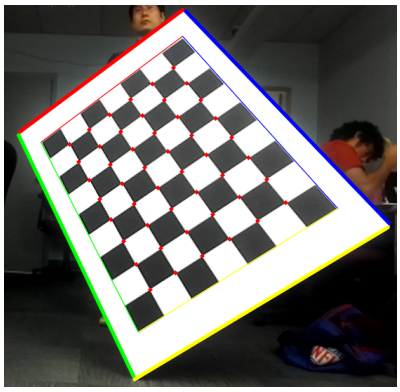

Fig. 3. Planes and Lines from Camera

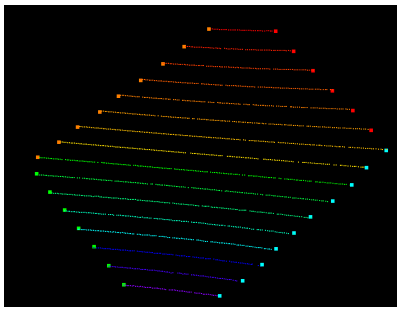

Fig. 4. Planes and Lines from 3D LiDAR

2) Planes and Lines Extraction from 3D LiDAR: For 3D LiDAR, we use user-provided cuboid to help locate the checkerboard in the point cloud. The 3D plane parameters can be computed by using the RANSAC algorithm to extract points on the checkerboard plane. Then we can filter points on scanlines according to the length of the consecutive segments to get a refined point cloud of the plane.

After getting the point cloud of the plane, we apply line fitting on each scanline to filter out outliers by RANSAC. And we can obtain the left boundary and right boundary of the checkerboard from the starting points and ending points on the scan lines of the plane point cloud.

In the last step, we split the left boundary and right boundary into four boundaries from detecting the turning point of the boundary, and we use line fitting again to filter out outliers to get the final 3D line parameters. The example result of planes and lines extraction is shown in Fig. 4.

D. Extrinsic Calibration of a Stereo Camera and a 3D LiDAR

In this section, we will describe the geometry constraints and extrinsic calibration algorithm of a stereo camera and a 3D LiDAR.

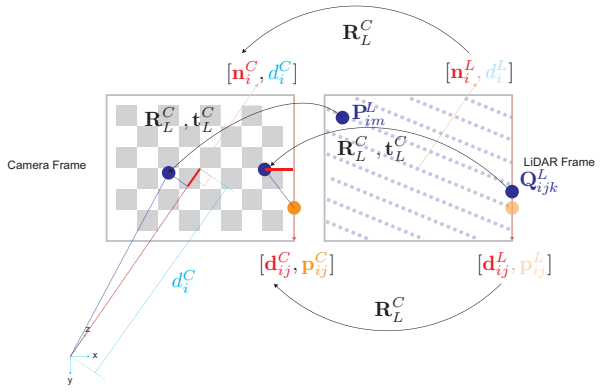

Fig. 5. Geometric Constraints

1) Geometric Constraints: The geometric constraints are illustrated in Fig. 5.

For line pairs $(L_{ij}^{C_1}, L_{ij}^{L})$ and $(L_{ij}^{C_2}, L_{ij}^{L})$, there are following constraints:

$$\mathbf{R}_L^{C_1}\mathbf{d}_{ij}^{L} = \mathbf{d}_{ij}^{C_1} \tag{9}$$

$$\mathbf{R}_L^{C_2}\mathbf{d}_{ij}^{L} = \mathbf{d}_{ij}^{C_2} \tag{10}$$

$$(\mathbf{I} - \mathbf{d}_{ij}^{C_1}(\mathbf{d}_{ij}^{C_1})^T)(\mathbf{R}_L^{C_1}\mathbf{Q}_{ijk}^{L} - \mathbf{p}_{ij}^{C_1} + \mathbf{t}_L^{C_1}) = \mathbf{0} \tag{11}$$

$$(\mathbf{I} - \mathbf{d}_{ij}^{C_2}(\mathbf{d}_{ij}^{C_2})^T)(\mathbf{R}_L^{C_2}\mathbf{Q}_{ijk}^{L} - \mathbf{p}_{ij}^{C_2} + \mathbf{t}_L^{C_2}) = \mathbf{0} \tag{12}$$

For plane pairs $(\pi_i^{C_1}, \pi_i^{L})$ and $(\pi_i^{C_1}, \pi_i^{L})$, there are following constraints:

$$\mathbf{R}_L^{C_1}\mathbf{n}_i^{L} = \mathbf{n}_i^{C_1} \tag{13}$$

$$\mathbf{R}_L^{C_2}\mathbf{n}_i^{L} = \mathbf{n}_i^{C_2} \tag{14}$$

$$\mathbf{n}_i^{C_1} \cdot (\mathbf{R}_L^{C_1}\mathbf{P}_{im}^{L} + \mathbf{t}_L^{C_1}) - d_i^{C_1} = 0 \tag{15}$$

$$\mathbf{n}_i^{C_2} \cdot (\mathbf{R}_L^{C_2}\mathbf{P}_{im}^{L} + \mathbf{t}_L^{C_2}) - d_i^{C_2} = 0 \tag{16}$$

In Eq. (9) (10), we use a rotation matrix between the LiDAR frame and the camera frame to transform the direction vector of the line from the LiDAR frame to the camera frame.

In Eq. (11) (12), we use rotation matrix and translation vector between the LiDAR frame and the camera frame to project $\mathbf{Q}_{ijk}^{L}$ which is one point on L_{ij}^{L} to the camera frame as $\mathbf{Q'}_{ijk}^{C}$, then connect it with $\mathbf{p}_{ij}$ to form the vector. And we use projection matrix $(\mathbf{I} - \mathbf{d}_{ij}^{C}(\mathbf{d}_{ij}^{C})^T)$ to project it to the vector from $\mathbf{Q'}_{ijk}^{C}$ to the line L_{ij}^{C}. This vector should have zero length since the projected point $\mathbf{Q'}_{ijk}^{C}$ should be also on the line L_{ij}^{C} in the camera frame.

In Eq. (13) (14), we use a rotation matrix between the LiDAR frame and the camera frame to transform the normal vector of the plane from the LiDAR frame to the camera frame.

In Eq. (15) (16), we use rotation matrix and translation vector between the LiDAR frame and the camera frame to project $\mathbf{P}_{im}^{L}$ which is one point on π_i^{L} to the camera frame as $\mathbf{P'}_{im}^{C}$, then project it on the normal vector $\mathbf{n}_i^{C}$ and minus d_i^{C} to get the distance to the plane π_i^{C}. Since the projected point $\mathbf{P'}_{im}^{C}$ should be also on the plane π_i^{C} in the camera frame, that distance to the plane π_i^{C} should have zero length.

2) Extrinsic Calibration from one pose: Using the above geometric constraints, [13] proves that we can compute the relationship between a 3D LiDAR and a monocular camera using 2 pairs of non-parallel line correspondences and 1 pair of plane correspondence with one pose. For the extrinsic calibration of a stereo visual system and 3D LiDAR, we can calibrate the 3D LiDAR with each camera to get the extrinsic parameters of the whole system.

3) Extrinsic Calibration from multiple poses: We can get a more accurate result using multiple poses. [13] introduced the extrinsic calibration method between a camera and a 3D liDAR from multiple poses using non-linear optimization. We can also formulate our problem as a non-linear optimization problem by minimizing line reprojection error and plane reprojection error from the 3D liDAR to the 3D camera to refine the extrinsic calibration result. And we can utilize LiDAR disparity constraint obtained by stereo calibration of the stereo visual system to optimize the extrinsic parameters globally.

Firstly, we minimize the following cost function to solve the initial $\mathbf{R}_L^{C_1}$ and $\mathbf{R}_L^{C_2}$:

$$\mathbf{R}_L^{C_1} = \arg\min_{\mathbf{R}_L^{C_1}} \sum_{i=1}^{N}\sum_{j=1}^{4} \|\mathbf{R}_L^{C_1}\mathbf{d}_{ij}^{L} - \mathbf{d}_{ij}^{C_1}\|^2 + \|\mathbf{R}_L^{C_1}\mathbf{n}_i^{L} - \mathbf{n}_i^{C_1}\|^2 \tag{17}$$

$$\mathbf{R}_L^{C_2} = \arg\min_{\mathbf{R}_L^{C_2}} \sum_{i=1}^{N}\sum_{j=1}^{4} \|\mathbf{R}_L^{C_2}\mathbf{d}_{ij}^{L} - \mathbf{d}_{ij}^{C_2}\|^2 + \|\mathbf{R}_L^{C_2}\mathbf{n}_i^{L} - \mathbf{n}_i^{C_2}\|^2 \tag{18}$$

As described in [28], the above problem can be solved by Singular Value Decomposition(SVD). We can define:

$$\mathbf{M}^{L} = [\mathbf{n}_1^{L}, \mathbf{d}_{11}^{L}, \ldots, \mathbf{d}_{14}^{L}, \ldots, \mathbf{n}_N^{L}, \mathbf{d}_{N1}^{L}, \ldots, \mathbf{d}_{N4}^{L}] \tag{19}$$

$$\mathbf{M}^{C_1} = [\mathbf{n}_1^{C_1}, \mathbf{d}_{11}^{C_1}, \ldots, \mathbf{d}_{14}^{C_1}, \ldots, \mathbf{n}_N^{C_1}, \mathbf{d}_{N1}^{C_1}, \ldots, \mathbf{d}_{N4}^{C_1}] \tag{20}$$

$$\mathbf{M}^{C_2} = [\mathbf{n}_1^{C_2}, \mathbf{d}_{11}^{C_2}, \ldots, \mathbf{d}_{14}^{C_2}, \ldots, \mathbf{n}_N^{C_2}, \mathbf{d}_{N1}^{C_2}, \ldots, \mathbf{d}_{N4}^{C_2}] \tag{21}$$

Using SVD, we can get

$$M^{L}(M^{C_1})^T = U_1\Sigma_1V_1^T \tag{22}$$

$$M^{L}(M^{C_2})^T = U_2\Sigma_2V_2^T. \tag{23}$$

Then, the rotation matrix $R_L^{C_1}$ and $R_L^{C_2}$ can be solved by closed-form solution:

$$R_L^{C_1} = V_1U_1^T \tag{24}$$

$$R_L^{C_2} = V_2U_2^T \tag{25}$$

To get intial translation vector $t_L^{C_1}$ and $t_L^{C_2}$, we need to utilize Eq. (11) (12) (15) (16). We denote $\bar{\mathbf{P}}_i^{L}$ as the centroids of the planes $\mathbf{P}_i^{L}$ in the LiDAR frame, $\bar{\mathbf{Q}}_{ij}^{L}$ as the entroids of the planes $\mathbf{Q}_{ij}^{L}$ in the LiDAR frame. Using above definitions and constraints, we can have

$$\mathbf{n}_i^{C_1} \cdot \mathbf{t}_L^{C_1} = -\mathbf{n}_i^{C_1} \cdot \mathbf{R}_L^{C_1} \bar{\mathbf{P}}_i^L + d_i^{C_1} \tag{26}$$

$$\mathbf{n}_i^{C_2} \cdot \mathbf{t}_L^{C_2} = -\mathbf{n}_i^{C_2} \cdot \mathbf{R}_L^{C_2} \bar{\mathbf{P}}_i^L + d_i^{C_2} \tag{27}$$

$$\mathbf{A}_{ij}^{C_1} \mathbf{t}_L^{C_1} = -\mathbf{A}_{ij}^{C_1} (\mathbf{R}_L^{C_1} \bar{\mathbf{Q}}_{ij}^L - \mathbf{p}_{ij}^{C_1}) \tag{28}$$

$$\mathbf{A}_{ij}^{C_2} \mathbf{t}_L^{C_2} = -\mathbf{A}_{ij}^{C_2} (\mathbf{R}_L^{C_2} \bar{\mathbf{Q}}_{ij}^L - \mathbf{p}_{ij}^{C_2}) \tag{29}$$

We can solve this least-square problem to get a closed-form solution of $\mathbf{t}_L^{C_1}$ and $\mathbf{t}_L^{C_2}$.

After having the initial rotation matrices $\mathbf{R}_L^{C_1}$, $\mathbf{R}_L^{C_2}$ and the translation vectors $\mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}$, we can solve a non-linear optimzation problem by Levenberg-Marquardt algorithm[26].

Considering Eq. (11) (12) (15) (16) as constraints, we can formulate following cost function terms which describe the constraits between 3D LiDAR and the cameras:

$$\begin{aligned}\mathbf{e}_L^{C_1} = &\sum_{i=1}^{N} \frac{1}{N_i} \|\mathbf{n}_i^{C_1} \cdot (\mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}) - d_i^{C_1}\|^2 \\ &+ \sum_{i=1}^{N} \sum_{j=1}^{4} \frac{1}{K_{ij}} \sum_{k=1}^{K_{ij}} \|\mathbf{A}_{ij}^{C_1} (\mathbf{R}_L^{C_1} \mathbf{Q}_{ijk}^L - \mathbf{p}_{ij}^{C_1} + \mathbf{t}_L^{C_1})\|^2\end{aligned} \tag{30}$$

$$\begin{aligned}\mathbf{e}_L^{C_2} = &\sum_{i=1}^{N} \frac{1}{N_i} \|\mathbf{n}_i^{C_2} \cdot (\mathbf{R}_L^{C_2} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_2}) - d_i^{C_2}\|^2 \\ &+ \sum_{i=1}^{N} \sum_{j=1}^{4} \frac{1}{K_{ij}} \sum_{k=1}^{K_{ij}} \|\mathbf{A}_{ij}^{C_2} (\mathbf{R}_L^{C_2} \mathbf{Q}_{ijk}^L - \mathbf{p}_{ij}^{C_2} + \mathbf{t}_L^{C_2})\|^2\end{aligned} \tag{31}$$

We jointly optimize two pairs of extrinsic calibration parameters between each camera and 3D LiDAR by minimizing the following cost function:

$$(\mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}) = \underset{\mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}}{\arg\min} \quad \mathbf{e}_L^{C_1} + \mathbf{e}_L^{C_2} \tag{32}$$

Extrinsic calibration of a stereo visual system has been explored by many researchers, we can obtain acure extrinsic parameters including rotation matrix and translation vector between two cameras using multiple images containing checkerboard. We adopt method [29] to calibrate two cameras to get accurate extrinsic parameters $(\mathbf{R}_{C_1}^{C_2}, \mathbf{t}_{C_1}^{C_2})$ of the stereo visual system.

Since the extrinsic calibration between a camera and a 3D LiDAR has estimation error, we can use LiDAR disparity constraint to consider this estimation error with stereo calibration result. This LiDAR disparity constraint is illustrated in Fig. 6. We project the point cloud of the checkerboard plane $\{\mathbf{P}_{im}^L\}$ in the LiDAR frame to the left camera frame as $\{\mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}\}$ using estimated $(\mathbf{R}_L^{C_1}, \mathbf{t}_L^{C_1})$. Then we project $\{\mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}\}$ to right camera frame as $\{\mathbf{R}_{C_1}^{C_2}(\mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}) + \mathbf{t}_{C_1}^{C_2}\}$using stereo extrinsic parameters $(\mathbf{R}_{C_1}^{C_2}, \mathbf{t}_{C_1}^{C_2})$ obtained by stereo camera calibration. This projected point cloud has disparity compared to $\{\mathbf{R}_L^{C_2} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_2}\}$ projected from LiDAR frame to right camera frame using $(\mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_2})$. This LiDAR disparity can be described as

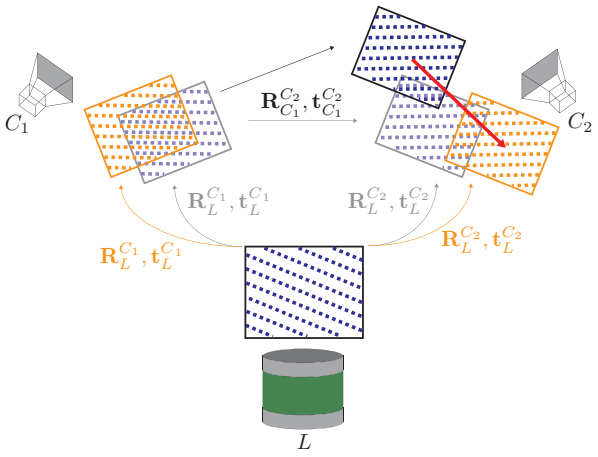

Fig. 6. LiDAR disparity constraint

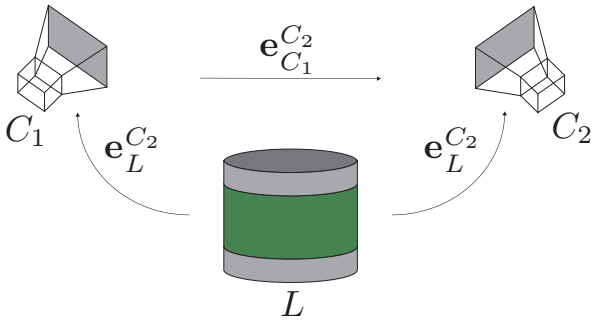

Fig. 7. Joint Optimization

$$\begin{aligned}\mathbf{e}_{C_1}^{C_2} = \sum_{i=1}^{N} \frac{1}{N_i} \sum_{m=1}^{N_i} &\|\mathbf{R}_{C_1}^{C_2}(\mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}) + \mathbf{t}_{C_1}^{C_2} \\ &- (\mathbf{R}_L^{C_2} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_2})\|^2\end{aligned} \tag{33}$$

Combining LiDAR disparity constraint with Eq. (11) (12) (15) (16) we can joinly optimize the extrinsic parameters by minimizing the following cost function which is also illustrated in Fig. 7:

$$(\mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}) = \underset{\mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}}{\arg\min} \quad \mathbf{e}_L^{C_1} + \mathbf{e}_L^{C_2} + \mathbf{e}_{C_2}^{C_1} \tag{34}$$

Since the measurement error of the grid size of the checkerboard will influence the extrinsic calibration result of a camera and 3D LiDAR, [13] introduced similarity transformation to replace the rigid transformation. We also apply similarity transformation to extrinsic calibration between a stereo visual system and 3D LiDAR.

We introduce a scale factor s to refine the scale factor for the laser beam of the 3D LiDAR which is used for getting actual distances of points. After getting initial $\mathbf{R}_\mathbf{L}^{\mathbf{C_1}}$ and $\mathbf{R}_\mathbf{L}^{\mathbf{C_2}}$

in the same way, we can solve the linear system which is showed as below to get initial s, $\mathbf{t}_L^{C_1}$ and $\mathbf{t}_L^{C_2}$.

$$\mathbf{n}_i^{C_1} \cdot \mathbf{t}_L^{C_1} + \mathbf{n}_i^{C_1} \cdot \mathbf{R}_L^{C_1} \bar{\mathbf{P}}_i^L s = -d_i^{C_1} \tag{35}$$

$$\mathbf{n}_i^{C_2} \cdot \mathbf{t}_L^{C_2} + \mathbf{n}_i^{C_2} \cdot \mathbf{R}_L^{C_2} \bar{\mathbf{P}}_i^L s = -d_i^{C_2} \tag{36}$$

$$\mathbf{A}_{ij}^{C_1} \mathbf{t}_L^{C_1} + \mathbf{A}_{ij}^{C_1} \mathbf{R}_L^{C_1} \bar{\mathbf{Q}}_{ij}^L s = \mathbf{A}_{ij}^{C_1} \mathbf{p}_{ij}^{C_1} \tag{37}$$

$$\mathbf{A}_{ij}^{C_2} \mathbf{t}_L^{C_2} + \mathbf{A}_{ij}^{C_2} \mathbf{R}_L^{C_2} \bar{\mathbf{Q}}_{ij}^L s = \mathbf{A}_{ij}^{C_2} \mathbf{p}_{ij}^{C_2} \tag{38}$$

To refine the result, the cost function without considering the LiDAR disparity constraint is showed as below:

$$\begin{aligned}
&(s, \mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}) = \underset{\mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}}{\arg\min} \\
&\sum_{i=1}^{N} \frac{1}{N_i} \| \mathbf{n}_i^{C_1} \cdot (s \mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}) - d_i^{C_1} \|^2 \\
&+ \sum_{i=1}^{N} \sum_{j=1}^{4} \frac{1}{K_{ij}} \sum_{k=1}^{K_{ij}} \| \mathbf{A}_{ij}^{C_1} (s \mathbf{R}_L^{C_1} \mathbf{Q}_{ijk}^L - \mathbf{p}_{ij}^{C_1} + \mathbf{t}_L^{C_1}) \|^2 \\
&+ \sum_{i=1}^{N} \frac{1}{N_i} \| \mathbf{n}_i^{C_2} \cdot (s \mathbf{R}_L^{C_2} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_2}) - d_i^{C_2} \|^2 \\
&+ \sum_{i=1}^{N} \sum_{j=1}^{4} \frac{1}{K_{ij}} \sum_{k=1}^{K_{ij}} \| \mathbf{A}_{ij}^{C_2} (s \mathbf{R}_L^{C_2} \mathbf{Q}_{ijk}^L - \mathbf{p}_{ij}^{C_2} + \mathbf{t}_L^{C_2}) \|^2
\end{aligned} \tag{39}$$

the cost function considering LiDAR disparity constraint is showed as below:

$$\begin{aligned}
&(s, \mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}) = \underset{\mathbf{R}_L^{C_1}, \mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_1}, \mathbf{t}_L^{C_2}}{\arg\min} \\
&\sum_{i=1}^{N} \frac{1}{N_i} \| \mathbf{n}_i^{C_1} \cdot (s \mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}) - d_i^{C_1} \|^2 \\
&+ \sum_{i=1}^{N} \sum_{j=1}^{4} \frac{1}{K_{ij}} \sum_{k=1}^{K_{ij}} \| \mathbf{A}_{ij}^{C_1} (s \mathbf{R}_L^{C_1} \mathbf{Q}_{ijk}^L - \mathbf{p}_{ij}^{C_1} + \mathbf{t}_L^{C_1}) \|^2 \\
&+ \sum_{i=1}^{N} \frac{1}{N_i} \| \mathbf{n}_i^{C_2} \cdot (s \mathbf{R}_L^{C_2} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_2}) - d_i^{C_2} \|^2 \\
&+ \sum_{i=1}^{N} \sum_{j=1}^{4} \frac{1}{K_{ij}} \sum_{k=1}^{K_{ij}} \| \mathbf{A}_{ij}^{C_2} (s \mathbf{R}_L^{C_2} \mathbf{Q}_{ijk}^L - \mathbf{p}_{ij}^{C_2} + \mathbf{t}_L^{C_2}) \|^2 \\
&+ \sum_{i=1}^{N} \frac{1}{N_i} \sum_{m=1}^{N_i} \| \mathbf{R}_{C_1}^{C_2} (s \mathbf{R}_L^{C_1} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_1}) + \mathbf{t}_{C_1}^{C_2} \\
&- (s \mathbf{R}_L^{C_2} \mathbf{P}_{im}^L + \mathbf{t}_L^{C_2}) \|^2
\end{aligned} \tag{40}$$

Then we also jointly optimize the extrinsic parameters by minimizing the cost function using the Levenberg-Marquardt algorithm to refine the result.

IV. Experiments and Results

To evaluate our method, we compare our method to Unnikrishnan's method [9] which only used the plane information to calibrate a camera and a 3D LiDAR. Our experiment uses a Velodyne VLP-16 LiDAR (16 scan lines, $\pm 3cm$ range error, 360° horizontal field of view and ±15° vertical field of view)and a ZED stereo camera(1280 × 720 resolution) to evaluate our method.

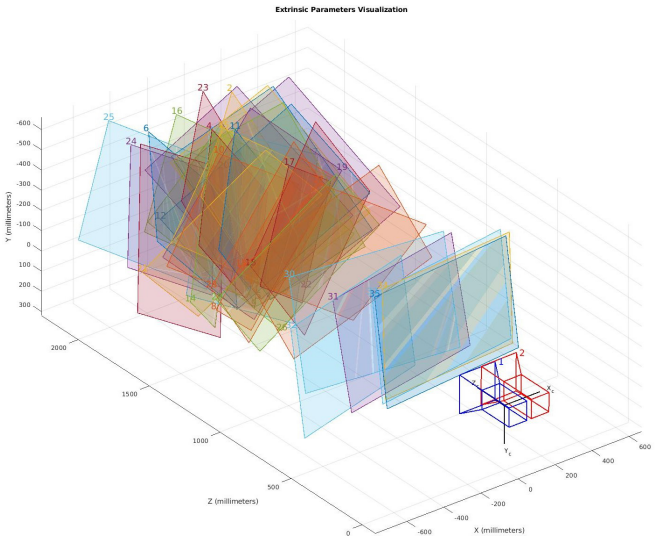

Fig. 8. Poses of checkerboards for the Stereo Calibration

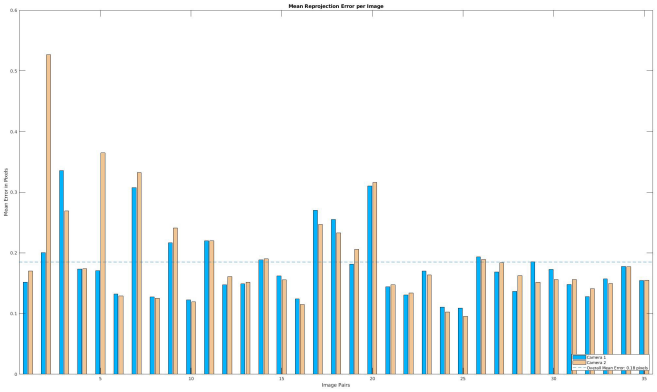

Fig. 9. Reprojection Error of the Stereo Calibration

We use stereo calibration with 35 poses to get accurate extrinsic parameters $(\mathbf{R}_{C_1}^{C_2}, \mathbf{t}_{C_1}^{C_2})$ of the stereo visual system. The poses of the checkerboards are showed in Fig. 8. The reprojection error of the stereo calibration is shown in Fig. 9.

To evaluate methods, we estimate the extrinsic parameters between each camera and 3D LiDAR $(\mathbf{R}_L^{C_1}, \mathbf{t}_L^{C_1}), (\mathbf{R}_L^{C_2}, \mathbf{t}_L^{C_2})$, then calculate the relative pose from left camera frame to right camera frame using these two transformation. For similarity transformation, we will regard it as the rigid transformation when we calculate the error.

To evaluate the transformation error between the estimated transformation $(\hat{\mathbf{R}}, \hat{\mathbf{t}})$ and the groundtruth transformation $(\mathbf{R}, \mathbf{t})$, we use $\frac{\|\hat{t}-t\|_2}{\|t\|_2}$ as the translation error and the angle-axis representation of $\hat{\mathbf{R}}\mathbf{R}^{-1}$ [30] as the rotation error.

We collected 30 pairs of LiDAR point clouds and images

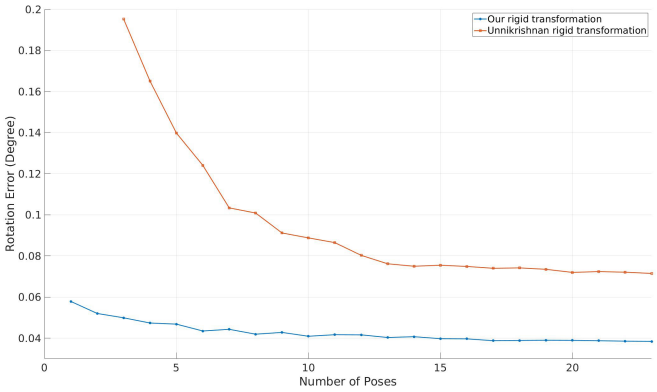

Fig. 10. Mean Rotation Error of rigid transformation, compare between our method without LiDAR disparity constraint and Unnikrishnn's algorithm[9]

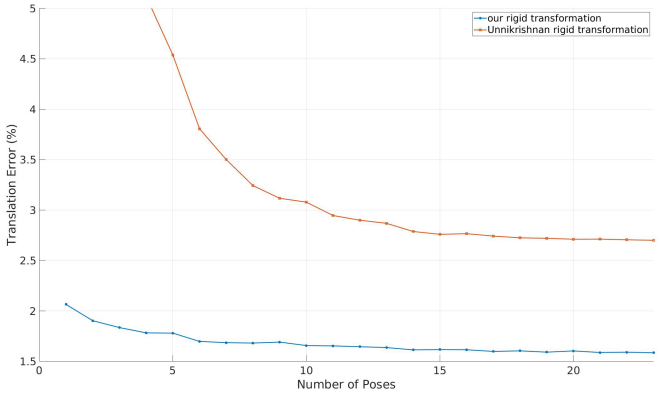

Fig. 11. Mean Translation Error of rigid transformation, compare between our method without LiDAR disparity constraint and Unnikrishnn's algorithm[9]

to do our evaluation, we choose $N \in [1, 23]$ randomly from them for our method, and $N \in [3, 23]$ for Unnikrishnan's method. We run the experiment 200 times for each $N \in [2, 25]$. We estimate extrinsic parameters for all 30 poses for $N = 1$.

The comparison of our method and Unnikrishnan's method is shown in Fig. 10 and Fig. 11. In this comparison experiment, we both use rigid transformation without using LiDAR disparity constraint to calibrate each camera with 3D LiDAR. The results show that our method significantly outperforms Unnikrishnan's method. And the results prove that our method can provide accurate extrinsic calibration result even with one pose.

We compare the result of rigid transformation and similarity transformation in different ways. Firstly, we use one pose to calibrate each camera with 3D LiDAR using rigid transformation and similarity transformation without using LiDAR disparity constraint. Secondly, we back-project the point cloud of the checkerboard plane from the LiDAR frame to the camera frame. The fusion result is showed in Fig. 12. We can observe from the fusion result that the similarity transformation provides a more accurate result that the rigid transformation.

We also show that the LiDAR disparity constraint can improve the extrinsic calibration result significantly by comparing the performance of rigid/similarity transformation with/without LiDAR disparity constraint. And the result shows that similarity transformation gives more improvement in translation estimation than rotation estimation. The rotation error and translation error is showed in Fig. 13 and Fig. 14.

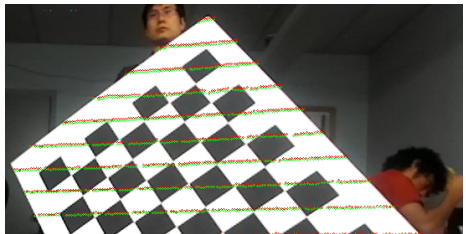

Fig. 12. Back-projection of the point cloud of the checkerboard plane using extrinsic parameters obtained from our rigid transformation method without LiDAR disparity constraint(red) and our similarity transformation method without LiDAR disparity constraint(green). Both methods use one pose.

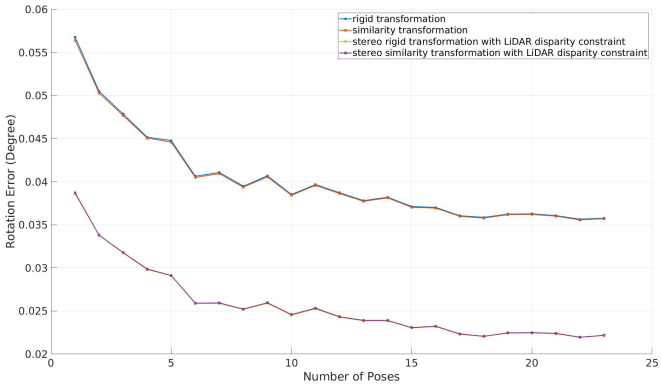

Fig. 13. Mean Rotation Error of our method: rigid/similarity transformation with/without LiDAR disparity constraint

V. Conclusions and Future Works

Using line and plane correspondences, we can use fewer (even one) poses to get more accurate extrinsic calibration result of a stereo visual system and 3D LiDAR. Using LiDAR disparity constraint, we can improve results with multiple poses significantly. We show that similarity transformation which does not require measurement of grid size of checkerboard can have more accurate result than rigid

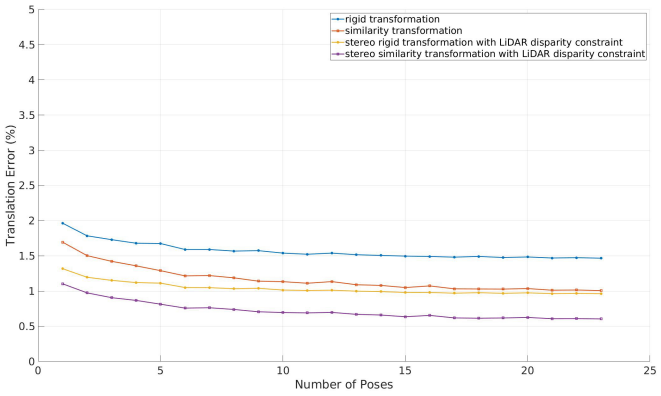

Fig. 14. Mean Translation Error of our method: rigid/similarity transformation with/without LiDAR disparity constraint

transformation. We implemented a calibration toolbox in C++ with high efficiency and accuracy. In the future, We will extend the calibration framework to multi-sensor calibration problem including intrinsic and extrinsic calibration.

ACKNOWLEDGMENT

Thanks to my mentor Lipu Zhou, Michael Kaess for guidance. Thanks to ShanghaiTech University for funding this work. Special Thanks to Dr. John Dolan, Ms. Rachel Burcin and the RISS team for support.

REFERENCES

[1] J. Levinson and S. Thrun, "Automatic online calibration of cameras and lasers." in *Robotics: Science and Systems*, vol. 2, 2013.

[2] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, "Automatic targetless extrinsic calibration of a 3d lidar and camera by maximizing mutual information." in *AAAI*, 2012.

[3] Z. Taylor and J. Nieto, "Automatic calibration of lidar and camera images using normalized mutual information," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013.

[4] T. Scott, A. A. Morye, P. Piniés, L. M. Paz, I. Posner, and P. Newman, "Choosing a time and place for calibration of lidar-camera systems," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4349–4356.

[5] X. Gong, Y. Lin, and J. Liu, "3d lidar-camera extrinsic calibration using an arbitrary trihedron," *Sensors*, vol. 13, no. 2, pp. 1902–1918, 2013.

[6] V.-D. Hoang, D. C. Hernandez, H.-S. Park, and K.-H. Jo, "Closed-form solution to 3d points for estimating extrinsic parameters of camera and laser sensor," in *Industrial Electronics (ISIE), 2014 IEEE 23rd International Symposium on*. IEEE, 2014, pp. 1932–1937.

[7] K. Kwak, D. F. Huber, H. Badino, and T. Kanade, "Extrinsic calibration of a single line scanning lidar and a camera," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 3283–3289.

[8] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[9] R. Unnikrishnan and M. Hebert, "Fast extrinsic calibration of a laser rangefinder to a camera," *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-05-09*, 2005.

[10] A. Geiger, F. Moosmann, Ö. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3936–3943.

[11] G. Pandey, J. McBride, S. Savarese, and R. Eustice, "Extrinsic calibration of a 3d laser scanner and an omnidirectional camera," in *7th IFAC symposium on intelligent autonomous vehicles*, vol. 7, no. 1, 2010.

[12] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[13] L. Zhou, Z. Li, and M. Kaess, "Automatic extrinsic calibration of a camera and a 3d lidar using line and plane correspondences," in *Intelligent Robots and Systems, 2018. IROS 2018. IEEE/RSJ International Conference on*. IEEE, 2018.

[14] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, "Automatic extrinsic calibration of vision and lidar by maximizing mutual information," *Journal of Field Robotics*, vol. 32, no. 5, pp. 696–722, 2015.

[15] D. Scaramuzza, A. Harati, and R. Siegwart, "Extrinsic self calibration of a camera and a 3d laser range finder from natural scenes," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 4164–4169.

[16] Y. Zheng, Y. Kuang, S. Sugimoto, K. Astrom, and M. Okutomi, "Revisiting the pnp problem: A fast, general and optimal solution," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2344–2351.

[17] N. Schneider, F. Piewak, C. Stiller, and U. Franke, "Regnet: Multimodal sensor registration using deep neural networks," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE, 2017, pp. 1803–1810.

[18] H. Aliakbarpour, P. Nunez, J. A. Prado, K. Khoshhal, and J. Dias, "An efficient algorithm for extrinsic calibration between a 3d laser range finder and a stereo camera for surveillance." in *ICAR*, 2009, pp. 1–6.

[19] C. Guindel, J. Beltrán, D. Martín, and F. García, "Automatic extrinsic calibration for lidar-stereo vehicle sensor setups," in *Intelligent Transportation Systems (ITSC), 2017 IEEE 20th International Conference on*. IEEE, 2017, pp. 1–6.

[20] Y. Li, Y. Ruichek, and C. Cappelle, "3d triangulation based extrinsic calibration between a stereo vision system and a lidar," in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 2011, pp. 797–802.

[21] V. John, Q. Long, Z. Liu, and S. Mita, "Automatic calibration and registration of lidar and stereo camera without calibration objects," in *Vehicular Electronics and Safety (ICVES), 2015 IEEE International Conference on*. IEEE, 2015, pp. 231–237.

[22] T. Scott, A. A. Morye, P. Piniés, L. M. Paz, I. Posner, and P. Newman, "Exploiting known unknowns: Scene induced cross-calibration of lidar-stereo systems," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 3647–3653.

[23] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[24] G. Bradski and A. Kaehler, "Opencv," *Dr. Dobbs journal of software tools*, 2000.

[25] K.-E. Jaeger, S. Ransac, B. W. Dijkstra, C. Colson, M. van Heuvel, and O. Misset, "Bacterial lipases," *FEMS microbiology reviews*, vol. 15, no. 1, pp. 29–63, 1994.

[26] J. J. Moré, "The levenberg-marquardt algorithm: implementation and theory," in *Numerical analysis*. Springer, 1978, pp. 105–116.

[27] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "Lsd: A fast line segment detector with a false detection control," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 4, pp. 722–732, 2010.

[28] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 5, pp. 698–700, 1987.

[29] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[30] Z. Kukelova, J. Heller, and A. Fitzgibbon, "Efficient intersection of three quadrics and applications in computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1799–1808.